Using Conceptual Models in Research Methods Courses: An experience using iStar 2.0

Marcela Ruiz, Fatma Başak Aydemir, and Fabiano Dalpiaz

Department of Information and Computing Sciences
Utrecht University, the Netherlands
`{m.ruiz, f.b.aydemir, f.dalpiaz}@uu.nl`

Abstract. Conceptual modelling languages are typically taught in graduate and postgraduate information systems programs. In the specific case of postgraduate information systems students, the lecturer has the opportunity to exploit conceptual modelling for educational purposes that go beyond learning how to apply the modelling paradigm or specific languages. In this paper, we report on our employment of the recently proposed iStar 2.0 in the master-level course Advanced Research Methods at Utrecht University in the Netherlands. During this course, the students conducted a design science project where iStar 2.0 artefacts are evaluated in an experimental setting. We present the course (intended learning objectives, learning materials, etc.), we explain how we employed iStar 2.0 therein, and we discuss our teaching experience including lessons learnt.

Keywords: iStar 2.0, education, research methods, experimental design

1 Introduction

Teaching conceptual modelling poses major challenges for the current educational programs in information and computer sciences. In this context, the lecture room lets the students learn novel conceptual modelling languages, use tools and techniques for the specification of conceptual models, apprehend guidelines and algorithms for transforming conceptual models into software systems, and come across methods that facilitate the practical application and evolution of conceptual modelling languages [1]. The variety of artefacts to be taught in courses related to conceptual modelling poses an interesting teaching dilemma: what are the appropriate methods for providing education about each artefact to bachelor and master students?

Given the complexity of the theoretical concepts related to conceptual modelling, it is common practice in higher education to teach business process modelling, databases, and software engineering in bachelor courses; while keeping goal analysis, meta-modelling, and enterprise modelling for master-level or advanced bachelor courses. Furthermore, students are taught how to construct models using a conceptual modelling language, but are hardly educated to reflect on the theoretical and practical challenges that researchers encounter when building a language or related artefact. This may create the false myth that conceptual modelling research is only about creating models.

For goal modelling, and particularly for the iStar framework [2] different experiences have been reported for bachelor [3], master [4], and mixed courses [5] [6]. The variety of iStar artefacts (methods, modelling tools, transformation guidelines from iStar models from/to other conceptual models, etc.) offers the opportunity to establish different settings where the iStar framework can be taught, exploited, and studied [7].

The Advanced Research Methods (ARM) course is a master-level course offered within the Department of Information and Computing Sciences at Utrecht University, the Netherlands, which is attended by circa 60 students per academic year. The first author of this paper redesigned this course for the period 2016-2017 by incorporating the Design Science method as a main component of the course [8]. Design Science is a research method for the design and investigation of artefacts in context, and prescribe engineering and empirical cycles for information systems projects in general.

For the ARM course, the students were made aware of research challenges concerning goal modelling; they conducted a design science project with a basic design cycle and a full empirical cycle, in which 4 existent iStar 2.0 [9] artefacts were investigated in an experimental setting. During the ARM course, the students took the roles of researchers and experimental subjects. Thus, we repot on our teaching experience where we employed conceptual modelling artefacts in a research methods course; and not in the conventional information systems and requirements engineering courses where the focus is on model building. Through this paper, we make the following contributions:

- Promoting the use of conceptual modelling in teaching environments.
- Reporting on the design of four comparative experiments in which existent iStar 2.0 artefacts have been used as experimental objects.
- Applying the design science method for the evaluation of iStar 2.0 artefacts in classroom environments.
- Discussing the employment of iStar 2.0 as part of the teaching material for research method courses in information systems programmes.

Paper organisation. Section 2 presents the intended learning outcomes and activities for the advanced research methods course; Section 3 describes the employment of iStar 2.0 during the ARM course; and Section 4 discusses lessons learnt and teaching directions for adopting iStar in teaching environments.

2   Intended learning outcomes and activities for the ARM course

The Master's Programme in Business Informatics (MBI) at Utrecht University, Department of Information and Computing Sciences, is a research master with an integrative and multidisciplinary approach that trains future ICT Researchers, Analysts, and Entrepreneurs. ARM[1] is a compulsory course and one of the pillars of the MBI program. The main objective of the course is to train the students to apply research methods, and give the students the opportunity to play the role of researchers in information sciences.

---

[1] https://sites.google.com/view/arm16-17

2.1   Intended learning outcomes

During the lectures and laboratory sessions, students gain skills to conduct research projects, develop a researcher attitude, and acquire knowledge about research methods. The intended learning outcomes (ILOs) for the ARM course are listed in Table 1.

Table 1. Intended learning outcomes for the ARM course

The student *is able to*
1. Design research projects that involve two main activities: artefact design and artefact evaluation
2. Conduct a comparative experiment to evaluate artefacts in contexts
3. Apply statistical tools for data analysis
4. Write scientific papers to report on research results
The student *shows*
5. A communicative attitude in working together to establish an overall goal
6. Willingness to evaluate his/her colleagues by means of peer reviews activities
7. A proactive attitude to schedule research activities, set up research environments, and apply research ethics
8. Motivation to present and publish scientific results
The student *knows*
9. Key concepts of the Design Science methodology
10. Experimental research protocols
11. A road map of research methods: Observational case studies, Single-case mechanism experiments, technical action research, canonical action research
12. Notions on ethics in experimentation
13. Basic concepts for data analysis and interpretation (scoping, planning, operation)
14. Statistics: Descriptive, parametric and non-parametric tests, linear regression
15. Structures for research presentation & package

To contribute to the successful achievement of ILOs 1 and 2, we provided students with conceptual modelling artefacts that can be evaluated in a certain context. We selected these among the iStar 2.0 artefacts and defined the following learning outcomes:

a) Knows how iStar 2.0, and social modelling languages in general, relate to other conceptual and enterprise modelling languages;
b) Is able to comprehend existing iStar 2.0 models by having learned the language's syntax and semantics;
c) Is able to create simple iStar 2.0 models.

To achieve the learning outcomes a), b) and c), in the same spirit as [3], we designed one lecture based on the iStar 2.0 tutorial given at ER 2016 and practical sessions that contributed to an active learning environment. In the following sections, we describe the activities, details about the background of the students and teaching materials.

2.2 Activities

For the ARM course, we designed different activities for one academic period of 10 weeks. Fig. 1 illustrates the main activities that took place during the 2016-2017 instance of the course. We kick-started the course with an introductory lecture, in which we explained the main objectives and activities of the course. The same day, we asked the students to fill in a demographic questionnaire with the purpose to get more insights about their background and previous experience with research methods, requirements engineering, conceptual modelling, goal modelling, business process modelling, and design science. As a result, we found out that 53.7% of the students (29 out of 54 students) had experience with conceptual modelling thanks to bachelor courses like information sciences and data modelling. One student reported to have industrial experience with the application of conceptual modelling for software development.
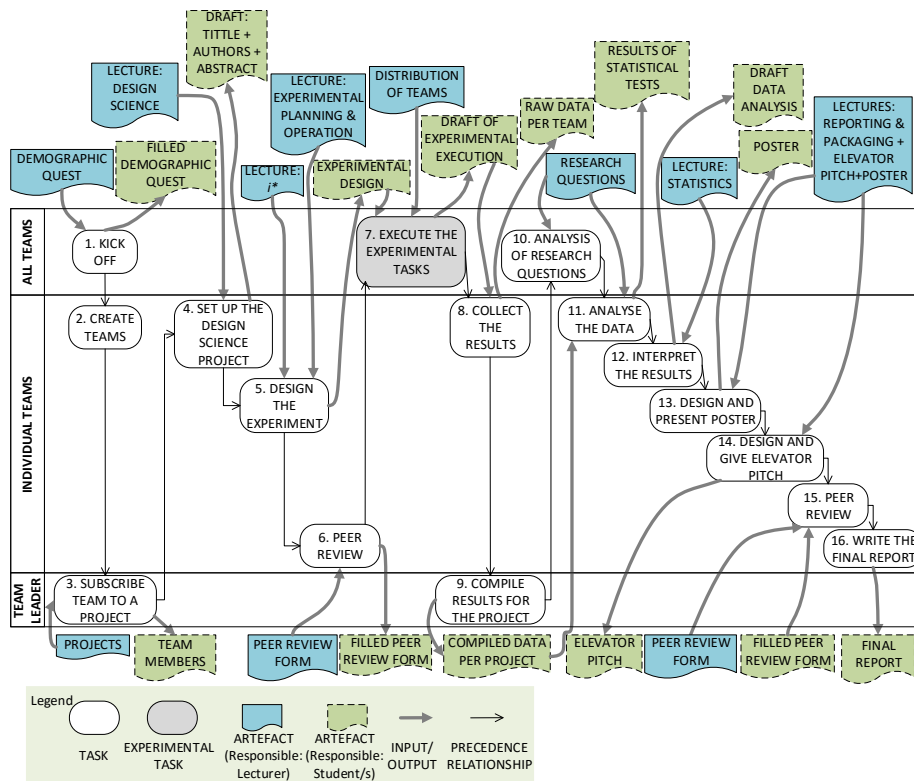

Fig. 1. Main activities for the ARM course

Regarding the syntactic knowledge level for goal modelling, 70.3% of the students reported a moderately low to average level of knowledge. On the contrary, 66.7% of the students rated themselves with an average to high level of experience for business process modelling. 22.2 % of the students had experience with experimentation in software engineering, and 24.1% experience with the design science method or received some

notions about it. Regarding their professional experience, 42.6% of the students have some professional experience; some roles to highlight are analyst, entrepreneur, programmer, and junior consultant.

For the next session (task 4 in Fig. 1), the students received the training to set up design science projects and started to study the selected artefact for each team. For the iStar lecture (task 5 in Fig. 1), the third author trained them to use iStar 2.0. For this, the students received an interactive lecture of 2 hours plus 2 hours of practice. For the practical assignment, the students took a scenario and modelled by using iStar. They were in charge of identifying the main actors, define their goals, find their dependencies, use intentional element links, analyse and evaluate alternative ways of fulfilling goals, create the models pen-on-paper, and scan and send the models the same day. We checked the models and the students received feedback to improve their models. The students were committed to the activity for two main reasons: 1) they wanted to learn the details about the artefact to evaluate, and 2) the models they prepared would serve as experimental objects for the comparative experiments (the main project of the course).

When the students finalised the experimental design and improved the iStar models, they conducted the experimental tasks. Finally, they collected data, performed data analysis, and reported the results via a scientific paper, a poster, and an elevator pitch.

3   Using iStar 2.0: the ARM course projects

During the ARM course, the students conducted a design science project with a basic design cycle and a full empirical cycle. For this, the students were distributed in teams of 3-4 people; each team elected a team leader (see Fig. 2). Since it was not the objective of the course to build information systems artefact from scratch, the students studied existent iStar 2.0 artefacts in the context of a design science project.

For the design cycle, each team selected 1 out of 4 iStar 2.0 artefacts. For this year, we have selected two artefacts that prescribe guidelines and techniques that use iStar 2.0 models (A1 and A2), and two artefacts that support the specification and syntactical notation of iStar 2.0 models (A3 and A4).

For the empirical cycle, each team conducted a comparative experiment for an iStar 2.0 artefact. During the empirical cycle, students analysed a given a problem with the artefact, designed an experiment, analysed the threats on the validity of the experiment, executed the experimental tasks, and analysed the experimental results. For the experimental tasks, the students acted as researchers of their own experiment, also as experimental subjects of the experiments of their researcher fellows. For example, in the case of the team A1.1, they took the role of researchers for the artefact P1 whereas they were the experimental subjects of the teams A2.1, A3.1, and A4.1 (see Fig. 2).
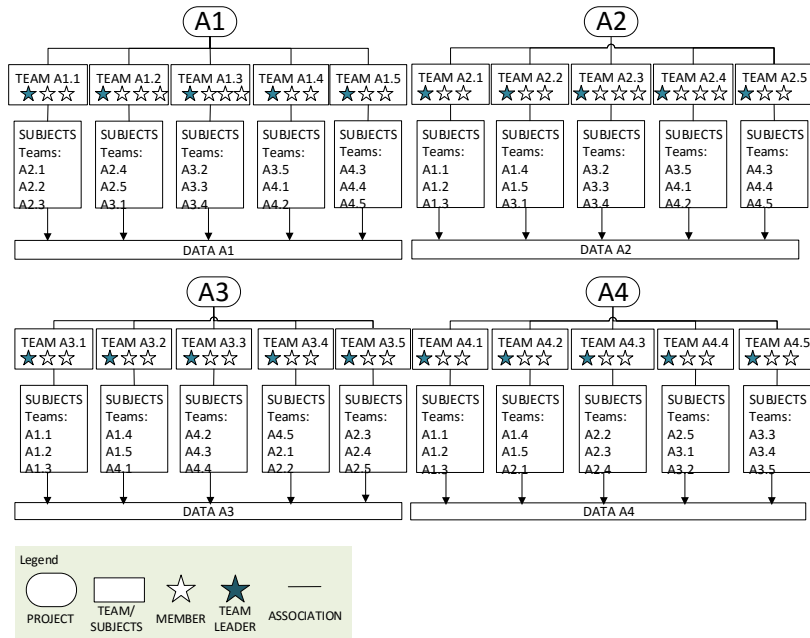
Fig. 2. Distribution of students per project, and assignation of experimental subjects

For the design of the comparative experiments, the students applied the experimental protocol for experimentation in software engineering prescribed by Wohlin et al. [10].

3.1 Experimental objects: iStar 2.0 artefacts

We provided the students with iStar 2.0 artefacts for their evaluation. Table 2 presents a summary of the artefacts[2].

Table 2. iStar 2.0 artefacts

Name	Description and general objective of the experiment
A1: iStar2ca guidelines [11]	**Description:** The GoBIS framework integrates two goal and business process modelling approaches: iStar and Communication Analysis (a communication-oriented business process modelling method) [12]. The GoBIS framework comprises The *iStar2ca* guidelines for a top-down scenario where its main purpose is to guide the mapping from iStar into Communication Analysis elements. **Objective:** Conduct a comparative experiment in order to evaluate the benefits and drawbacks of using the *iStar2ca* guidelines in terms of performance and perceptions from a practitioner point of view.

---

[2] Further details about the given artefacts are available as part of the course material: https://sites.google.com/view/arm16-17/material

A2: Delta Analysis technique [13]	**Description:** Delta Analysis is a technique for the analysis of differences and information gathering of two information systems. The Delta Analysis technique serves on the purpose to analyse the delta of two information systems. Delta Analysis is model-based; thus, the comparison or delta is performed between pair of models that specify information systems. The Delta Analysis technique is general enough to be applied to any pair of conceptual models (e.g., specification of information system goals, business process, interaction requirements, etc.). **Objective:** Conduct a comparative experiment in order to evaluate the benefits and drawbacks of using the Delta Analysis technique in terms of performance and perceptions from a practitioner point of view when comparing iStar models.
A3: piStar tool [14]	**Description**: Just like other modelling languages, iStar 2.0 modellers often make errors and create models that are not compliant with the syntax. To such extent, iStar 2.0 comes with meta-model enriched with additional constraints about syntactic well formedness. iStar 2.0 models can be drawn by hand on paper, digitally using a general purpose drawing tool such as Microsoft Visio, or using a dedicated application for iStar 2.0 such as piStar. **Objective:** Conduct a comparative experiment in order to evaluate the impact of using piStar in terms of performance and perceptions from a practitioner point of view.
A4: iStar 2.0 notation and Moody's visual notation [15] [16]	**Description**: The standard iStar 2.0 notation uses standard shapes (from the original version of iStar) to represent concepts. For example, circles represent actors, stadium shaped nodes represent goals, and hexagons represent tasks. The relationships between these constructs are captured by links between those shapes, sometimes labelled by text. The arrow heads used to connect the directed edges may differ for different relationships. Moody et al. discuss the advantages and disadvantages of the standard iStar notation and suggest improvements on the visual notation of iStar, basing these suggestions on theory of visual design. **Objective**: Conduct a comparative experiment in order to evaluate the benefits of using the Moody's notation for iStar 2.0 models in terms of performance and perceptions from a practitioner point of view

To illustrate the experimental setups used in the course, we describe the project of the students that have selected the artefact A4. In this project, the students (5 teams) have designed a comparative expeiment where the standard iStar2.0 notation is compared against the Moody's visual notation in terms of practioners' performance and perceptions (see Fig. 3). Each team had between 9 to 11 experimental subjects. For this experiment, we defined dependent and independent variables, which were measured by means of two experimental tasks in two weeks (see Fig. 3, week 1 and week 2). Each team of artefact A4 elected one context and specified models using the standard iStar 2.0 and Moody's notations. In the first experimental task the experimental subjects were divided in two groups: the group A analysed a goal model specified by means of the iStar 2.0 notation, and the group B analysed a goal model based on Moody's notation. In the second experimental task, each team chose a different context from the one used for the experimental task 1 and defined the iStar 2.0 and Moody's models. In week 2, the teams switched the experimental subjects as described in Fig. 3; and kept the same complexity of the experimental objects in terms of the amount of modelling elements and type of elements that were used in the models for the experimental task 1.
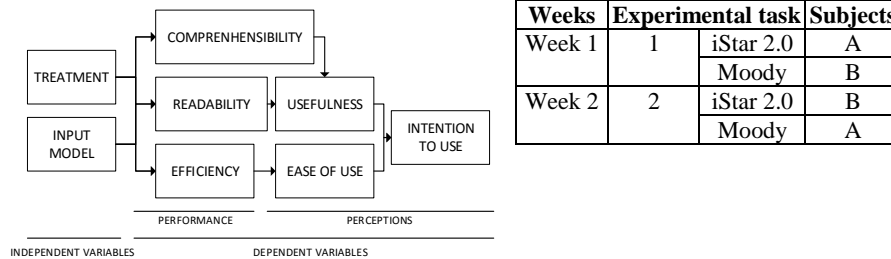
Weeks	Experimental task		Subjects
Week 1	1	iStar 2.0	A
		Moody	B
Week 2	2	iStar 2.0	B
		Moody	A

Fig. 3. Independent and dependent variables, and experimental setting of the artefact A4

To measure comprehensibility and readability, the teams have designed questionnaires with context dependent questions about the iStar 2.0 and Moody's models. In this case, 10 questions were dedicated to readability and 5 to comprehensibility. For the efficiency, each team registered the amount of time that each subject took to finish the questionnaire. Finally, the teams measured usefulness, ease of use and intention to use according to a quality framework [17].

4 Discussion and lessons learnt

In this paper, we reported our experience in using conceptual modelling as artefact for teaching non-conventional information systems courses like research methods. A key aim of our attempt is to make students understand that conceptual modelling research goes well beyond reading and creating models, but rather involves thorough theoretical and empirical studies concerning conceptual modelling artefacts.

In our case, we describe the design of the Advanced Research Methods course (ARM), which has as a main component the Design Science Method. As part of the main project of the course, students conducted a design science project in order to evaluate artefacts in context. For the academic year 2016-2017, students evaluated conceptual model artefacts by means of a comparative experiment, wrote a research paper, and presented the results by means of a poster and elevator pitch. As a proof of concept, the students evaluated iStar 2.0 artefacts.

Reflecting on the intended learning outcomes and our experience, we highlight the following positive findings and opportunities for improvement.

4.1 Positive findings

- *The existence of conceptual modelling language variants can be beneficial!* One of the main criticisms made to the conceptual modelling community is that many variants of a given language (and related artefacts) are proposed. However, this drawback turned into an advantage for the ARM course, for we could easily select different iStar 2.0 artefacts to evaluate. Other conceptual modelling languages, such as for business processes, offer a comparable artefact selection space to be exploited.

- *Easily customizable materials and artefacts.* Conceptual modelling artefacts are adaptable for classroom settings without investing considerable resources. It is relatively simple and nonintrusive to modify the graphical notation, alter the syntax, or build a new analysis algorithm. Compare this to the effort required to modify a software product (e.g., user interface, algorithm, input devices). Thus, conceptual models are excellent artefacts for use in a research methods course.

- *Experimental outcomes trigger new design cycles for conceptual model artefacts.* The students gained first-hand experience about the limitations of existing conceptual modeling artifacts. This is a much more powerful learning experience compared to a teacher *telling* them that a modelling language suffers from a certain problem. We are confident that the obtained findings will lead some students to follow a new design cycle during their master's thesis.

- *Comparable complexity of the experiments.* Despite the differences of the four evaluated artefacts, it was possible for the instructor to keep a balance in workload of each team in terms of artefacts to build and variables to measure. For example, the teams measured subject performance and perceptions for each artefact, which required them to build experimental objects (iStar 2.0 models plus other objects according to the type of artefact, like Moody's models), create questionnaires to evaluate subjects' perceptions, and record the time spent per subject during the execution of the experimental task. The balance in the design and workload of the projects was a key point to avoid frustration and competition among the teams.

4.2 Opportunities for improvement

- *Managing multiple experiments at the same time.* It was difficult to assist four different research projects and schedule the parallel execution of 40 experimental task (20 teams, two experimental tasks each) in two weeks. These aspects need to be readjusted by, e.g., offering only a couple projects and fewer experimental objects.

- *Testing the achievement of the iStar ILOs.* Although we supported students by giving them feedback about their models until they reached a solid version, we did not evaluated the actual knowledge in iStar modelling. For the next edition of the ARM course, it is necessary to evaluate the extent to which each student achieved the iStar-related ILOs; this will reduce the threats to validity of the experiments given the analysis of iStar models that was required for all the experimental tasks.

- *On the difficulty of establishing scientific rigor.* Students learnt how to identify threats to the validity of the experiments, but we did not have sufficient time or experience to avoid many of them. Some threats were related to the difficulty to manage four experiments and to ensure the participation of the students in the experimental tasks. We noticed that the students tend to feel frustrated if they need to confront a threat to the validity of their experiments. It is important to allocate sufficient time for the students to prevent the most important threats and to teach them how to mitigate the threats when they occur. For example, testing the subjects' knowledge of iStar would help ensure an appropriate execution of the experimental tasks.

Looking at the intended learning objectives for ARM, we find that the experimental setting and the employment of iStar 2.0 are appropriate for a master-level course, especially thanks to the existence of many artefacts and the possibility to follow a full experimental protocol. In the upcoming years, we plan to improve the course design by establishing a better distribution of teams and less artefacts to evaluate. Also, we plan to reduce the amount of experimental objects (one per artefact, but not one per team), and to help the students manage the threats to the validity of the experimental results.

5 References

1. Cheng, J. and I.-Y. Song. Preface to SCME (Symposium on Conceptual Modeling Education). in International Conference on Conceptual Modeling. 2013. Springer.
2. Yu, E., Modelling Strategic Relationships for Process Reengineering, in Department of Computer Science. 1995, University of Toronto.
3. Dalpiaz, F., Teaching Goal Modeling in Undergraduate Education, in International iStar Teaching Workshop. 2015: Stockholm, Sweden.
4. Ruiz, M., et al., GoBIS: An integrated framework to analyse the goal and business process perspectives in information systems. Information Systems, 2015. **53**: p. 330-345.
5. Horkoff, Observational studies of new i* users: challentes and recommendations, in International iStar Teaching Workshop. 2015: Stockholm, Sweden.
6. Svee, E.-O. and J. Zdravkovic, iStar Instructions in Mixed Student Cohort Environments, in International iStar Teaching Workshop. 2015: Stockholm, Sweden.
7. Horkoff, J., et al., Taking Goal Models Downstream: A Systematic Roadmap, in Reserach Chanllenges in Information Science. 2014.
8. Wieringa, R., Design Science Methodology for Information Systems and Software Engineering. 2014: Springer-Verlag Berlin Heidelberg.
9. Dalpiaz, F., X. Franch, and J. Horkoff, iStar 2.0 Language Guide. 2016.
10. Wohlin, C., et al., Experimentation in Software Engineering. 2012: Springer.
11. Ruiz, M., et al., GoBIS: An integrated framework to analyse the goal and business process perspectives in information systems. Information Systems Journal, 2015. **53**: p. 330-345.
12. España, S., A. González, and Ó. Pastor, Communication Analysis: A Requirements Engineering Method for Information Systems, in Advanced Information Systems Engineering. CAiSE 2009. 2009, Springer.
13. Ruiz, M., TraceME: Traceability-based Method for Conceptual Model Evolution, in Departamento de Sistemas Informáticos y Computación. 2016, Universitat Politècnica de València: Valencia, Spain.
14. Pimentel, J. piStar: *http://www.cin.ufpe.br/~jhcp/pistar/*. 2016 [cited 2017.
15. Moody, D., P. Heymans, and R. Matulevicius, Visual syntax does matter: improving the cognitive effectiveness of the i* visual notation. Requirements Engineering, 2010. **15**(2): p. 141-175.
16. Genon, N., et al., Towards a More Semantically Transparent i* Visual Syntax, in REFSQ'12 Proceedings of the 18th international conference on Requirements Engineering: foundation for software quality. 2012, Springer: Essen, Germany.
17. Moody, D., et al., Evaluating the Quality of Process Models: Empirical Testing of a Quality Framework, in International Conference on Conceptual Modeling (ER 2002). 2002.